THE SEED

December 1986
Vol. 8 No. 12

NEWSLETTER FOR APPLE PEOPLE

DENVER APPLE PI

Was I s'posed to bring a //c, //e, gs or Mac

denver apple pi Seed

P.O. BOX 17467
DENVER, COLORADO 80217

President .....Roy Dean
Vice President & Property .....Trent Hein
Secretary & Editor .....Randall Ansley
Treasurer .....Bob Hein
Board Member & Software Librarian .....Ed Thompson
Board Member & Store Rep .....Tom Angelos
Board Member .....Bill Cronister
Sales Officer .....Chris Saxon
BBS Officer .....Robert Racansky

MEMBERSHIP INFORMATION

APPLE PI is an Apple Computer Users Group - not affiliated with Apple Computer, Inc. - that provides many member benefits, some of which are:

MONTHLY MEETINGS - On the first Thursday of each month at 7:00 p.m. APPLE PI holds its general meeting at the Green Center, Colorado School of Mines Campus, Golden, Colorado. This meeting is a combination of an informational meeting, new product announcements, and a short business meeting. The members are given an opportunity to buy computer supplies a little above cost. Software products are given away as door prizes.

On the Sunday following the general meeting there is an informal meet which is usually held at the Clear Creek Office Park, 4251 Kipling, in Wheat Ridge. This meeting is from 10:00 a.m. until 3:00 p.m. and provides a more informal environment to seek or give advice to fellow members and get copies of programs from the APPLE PI Library. Both of these meetings are open to the general public but only members have access to the club library and products that are sold.

PROGRAM LIBRARY - APPLE PI currently has a library of 9,000 programs. All current disks are in 16 sector (3.3, Pascal, CP/M) format and are available to any member. APPLE PI Members can obtain copies from the library:

1) By coming to the monthly informal meeting (swap meet);
2) By written request to the club librarian (at any meeting or by mail);
3) By electronic mail on APPLE PI's bulletin board system:

DISK OF THE MONTH - Each month a collection of some of the best programs donated to APPLE PI during the preceding month are made available. The DOM can be purchased at any group meeting or by mail.

DUES - Current membership dues are **$18.00** per year, with a one-time application fee of **$10.00.** Checks should be made payable to DENVER APPLE PI.

The words APPLE, APPLE COMPUTER, and the APPLE LOGO are registered trademarks of the Apple Computer Company Inc. Representation of these trademarks in the Seed does not constitute any expression of agreement or disagreement by Apple Computer Inc. with any articles or other printed material found in the Seed. The opinions, policies and comments contained in the Seed are those of the Apple Pi users group, officers and members.

NOTICE

Articles, stories, and other items in the Seed may be copied or used by other clubs or groups as long as credit is given to Denver Apple Pi and the author of the article. On reprints please include the date or volume number from the Seed.

ADVERTISING RATES

1/4 Page	$ 25.00
1/2 Page	$ 40.00
Full Page	$ 70.00

CONFESSIONS OF A BASIC LOVER

PART II

Last month we saw a little about how Applesoft looked like internally, and typed in a program to list the Applesoft commands and error messages. This month we'll see how to customize the Applesoft command dictionary and create our own keywords.

To create custom Applesoft commands, we have to graft our commands into Applesoft. The Applesoft interpreter resides in ROM addresses $D000 to $F800 Hex, and we can't change ROM (Read Only Memory), by definition. To change Applesoft around, we first have to move Applesoft to Random Access Memory. Also, to have everything work as we would expect, we need the RAM at the same addresses as Applesoft ROM, Hex $D000 to $F800.

Early Apples provided user RAM from $0000 to $BFFF (48K), 4K of memory mapped I/O (Input/Output) from $C000 to $CFFF, language (Applesoft or Integer Basic) and monitor ROM from $D000 to $FFFF. Apple IIs and II+s provided for a 'language card' in slot 0, a 16K area of ROM or RAM that mapped over the motherboard mounted language ROMs from $D000 to $DFFF twice, (8K), plus the 8K from $E000 to $FFFF. In my first Apple, I had a language card with Applesoft and the early Apple Monitor in 2716 ROM's. The card had a little red switch at the back. If I flipped the switch up, Applesoft would be resident after a cold start, while the Integer Basic on the motherboard would be disabled. If the switch were left down, the Integer Basic > prompt would come up after booting the system.

In any case, it was possible to get a RAM card, an extra 16 K user memory mapped over the ROMs from $D000 to $FFFF Hex. The 4K area from $D000 to $DFFF is mapped twice in this scheme, since $D000 to $FFFF represents only 12 K (4K of memory for each of $D000 to $DFFF, $E000 to $EFFF, and $F000 to $FFFF), while RAM cards have 16K of memory.

When the Apple IIe came along, it came with 64K rather than 48K of user RAM. The IIe adopted the same conventions for using the upper 16K of RAM as had been established earlier, namely bank switching between ROM and RAM using 'soft switches'.

The way the Apple's harware switches beween ROM and RAM is through the use of memory locations that are accessed twice to set or clear harware enable lines to ROM or RAM. There are a number of combinations available. For example, it's possible to read from ROM and write to RAM in the same memory locations. The four low address bits when $C08X is accessed control the combination of ROM and RAM switched in, and whether the RAM is write protected or write enabled. Address bit 3 controls whether the first or second 4K bank or RAM is used. We will only use the bank two. The possibilities include:

A3	A2	A1	A0	
0	0	0	0	Read RAM, Write Protect RAM
0	0	0	1	Read ROM, Write to RAM
0	0	1	0	Read ROM, Write Protect RAM
0	0	1	1	Read RAM, Write to RAM

Thus:

$C080 reads RAM, write protects RAM
$C081 reads ROM writes to RAM
$C082 reads ROM, write protects RAM
$C083 reads RAM, writes to RAM

Casual switching beween ROM and RAM in a 64K Apple can lead to strange results. Let's say you're in the monitor ROM, and you activate the main 12K of RAM by accessing location $C083 twice. If nothing useful is in upper RAM when you switch it in, you will inevitably be sent somewhere in upper memory (now RAM) by either the monitor or the BASIC interpreter, and the system will hang. If the 6502 can't find any useful instruction when it is switched form ROM to RAM, nothing useful will happen.

Since we know we have to get Applesoft to RAM before we can change it, let's look at the ways we can make the transfer. The easiest way is to read ROM and write what

we read to RAM. A BASIC program to do this would first select the appropriate ROM/RAM softswitch, set it, and read ROM into RAM. The softswitch that reads ROM and writes RAM is $C081, and must be addressed twice.

Converting to decimal, $C081 is:

```
12 * 4096 added to
 0 *  256 added to
 8 *   16 added to
 1 *    1
   49281
```

A program to write Applesoft to RAM might read:

```
10 X = 12 * 4096 + 8 * 16 + 1
20 Y =  PEEK (X):Y =  PEEK (X)
25 A = 13 * 4096:B = 15 * 4096 +
    15 * 256 + 15 * 16 + 15
30  FOR I = A TO B:Y =  PEEK (I)
    : POKE I,Y: NEXT
```

In this program, line 20 sets the soft switch, while line 30 reads ROM and writes RAM. The FOR loop is optimized for speed by using variables as the range for I, using NEXT without a variable name at the end of the loop and, most importantly, keeping the FOR-NEXT loop all in the same line number. Lots of time is required to set the Applesoft interpreter up for a new line.

Alternately, we can store Applesoft on disk by typing BSAVE BASIC,A$D000,L$3000, and then load it into RAM (first setting softswitch $C081 again) by typing BLOAD BASIC,A$D000.

Bank-switched RAM (and the Applesoft interpreter we put there) can then be read from and written to by typing Y = PEEK (X+2) twice, setting the softswitch that reads RAM and writes RAM ($C083).

We can test to see that everything works by typing in a few Applesoft commands, such as FOR I=1 TO 10:PRINT I:NEXT I.

When we're sure that Applesoft is in RAM and is behaving normally, we can start to change it around. A relatively easy change to make is to change the command dictionary which starts at $D0D0. We listed the reserved words last time using the program:

```
10  START = (13 * 4096) + (0 * 256) +  (13
* 16) + (0 * 1)
20  FIN = START + 660
30   FOR I = START TO FIN
40  Y = PEEK (I)
50   PRINT CHR$ (Y);
60  NEXT I
```

If we actually look at the dictionary using the monitor, we'll see hex bytes representing the ASCII characters for each command with one subtlety: the last letter of each keyword in the dictionary has its high bit set.

We can use that fact to replace the original Applesoft commands with our own equivalents. The commands will still act the same, only their names will differ. Let's say we wanted to change the command END with one of our own choosing, say FIN. We simply replace the dictionary entry END with FIN at the right location and using the right format.

The Applesoft BASIC Programming Reference Manual put out by Apple is of some help here. On page 121 (Appendix F) there is a listing of Decimal Tokens For Keywords which lets us know that the token for END is 128 decimal or 80 Hex. The keywords appear in the dictionary in token order, so END is the first entry. If we poke the ASCII characters for FIN into the dictionary starting at $D0D0, and set the high bit of the last letter N, then we should get an error message if the interpreter sees the old command END.

in immediate mode we can initialize X to point to the start of the keyword dictionary by typing:

X=13*4096 + 13*16 (RETURN)

We can change the keyword to FIN by poking in the new values:

```
POKE X, ASC("F") (RET)

POKE X+1, ASC("I") (RET)

POKE X+2, ASC("N")+128 (RET)
```

Adding 128 to ASC("N") sets the high bit, as the keyword dictionary format requires.

If we write a program that FINs on a condition like:

```
10 For  I = 1 to 100
20 IF I > 10 THEN FIN
30 NEXT I
```

the program should terminate when I gets to 11. Alternately, if the word FIN is changed to the old END, we should get a syntax error in 20 when I = 11.

If a syntax error has occured, the soft switches may get reset, so it wouldn't hurt to switch in auxilliary RAM again by getting into the monitor with a CALL -151, and reading or writing C083 twice to set auxilliary RAM for both reading and writing.

Now that we can change one keyword and make it work, we can write a program to change them all. The program included here displays the old keywords and prompts the user for a new keyword. Both the address of the keyword dictionary entry and the old keyword are displayed on the screen.

The program first loads Applesoft into auxilliary RAM (lines 5 to 10). Most of the subroutine at line 60 is concerned with converting the keyword dictionary entry's address from decimal to hex and assembling the hex address into H$ for printing. A$ is the character string containing the old keyword which was assembled in line 30 and printed in line 120. The new keyword to replace the current keyword dictionary entry is requested form the user in a subroutine starting at line 200 and called from line 130.

```
 5  HOME :X = 12 *  4096 + 8 * 16 + 1:Y  =
PEEK (X):Y =  PEEK (X):X = (13 * 4096):Y =
15 * 4096 + 15 * 256 + 15 * 16 + 15
 10 : VTAB (10): HTAB (10): PRINT "LOADING
BASIC INTO RAM": FOR I = X TO Y:Z =   PEEK
(I): POKE I,Z: NEXT : HOME  :X = X + 13  *
16
 11 HR$ = "123456789ABCDEF"
 12 Y = 12 * 4096 +  8 * 16 + 3:Z =   PEEK
(Y):Z =  PEEK (Y)
 15  REM  $D0D0
 20 Y = X + 661
 25  REM  $ D364
 30 Z =  PEEK (X):A$ = A$ +  CHR$ (Z):  IF
Z > 128 THEN  GOSUB 60:A$ = ""
 40 X = X + 1: IF X > Y THEN  FIN
 50  GOTO 30
 60 W =   LEN (A$):L1 = (X  - W) + 1:H$  =
"":L2 = L1
 70 A = 4096:L3 =  INT (L2 / A):H$ = H$  +
MID$ (HR$,L3,1):L2 = L2 - (A * L3)
 75  IF L2 < 256 THEN H$ = H$ + "0":  GOTO
82
 80 A = 256:L3 =   INT (L2 / A):H$ = H$  +
MID$ (HR$,L3,1):L2 = L2 - (A * L3)
 82  REM
 85  IF L2 < 16  THEN H$ = H$ + "0":  GOTO
95
 90 A = 16:L3  =  INT (L2  / A):H$ = H$  +
MID$ (HR$,L3,1):L2 = L2 - (A * L3)
 95  IF  NOT L2  THEN H$ = H$ + "0":  GOTO
120
 100 :H$ = H$ +  MID$ (HR$,L2,1)
 120   VTAB (10):  PRINT   SPC( 39):  VTAB
(10): HTAB (01): PRINT H$;" ";A$;
 130  GOSUB 200: PRINT : RETURN
 200  HTAB  (20): INPUT "  ";N2$: IF   LEN
(N2$) >  LEN (A$) THEN N2$ =  LEFT$  (N2$,
LEN (A$))
 205  HTAB (21): IF N2$ = "" THEN  RETURN
 206 X8 = X - ( LEN (A$)):X8 = X8 + 1
 210 Z9 = 0: FOR I = X8 TO X:Z9 = Z9 + 1
 220  POKE I, ASC ( MID$ (N2$,Z9,1)): IF I
= X THEN  POKE I, ASC ( MID$ (N2$,Z9,1)) +
128
 230  NEXT :N2$ = ""
 240  RETURN
```

Any changes made to Applesoft can be saved by typing:

```
BSAVE BASIC, A$D000, L$3000 (RET)
```

The modified version of Applesoft can then be loaded by reading $C081 twice, thus read enabling ROM and write enabling bank-switched RAM, and typing BLOAD BASIC at the monitor. The new version of BASIC can be activated by reading $C083 twice to both write and read enable bank-switched RAM.

denver apple pi

BBS PHONE NUMBER
422 - 3716

HOT LINE
422 - 0005

Apple® II

Compatibility Guide

Hardware Products

Product	Company	IIe	IIc	IIGS	Comments
100 Robotic Modem	Visionary	X		X	
7710 Serial Card	Cal. Computer Systems	X		X	
Accelerator IIe	Titan Technologies	X		Z	
Apple IIc Memory Expansion Card	Apple Computer		X		
Apple IIc Monitor (Monochrome)	Apple Computer	X	X	X	
Apple IIe 80-Column Text Card	Apple Computer	X		Z	No AUX slot on IIGS
Apple IIe Ext. 80-Column Text Card	Apple Computer	X		Z	No AUX slot on IIGS
Apple IIe Memory Expansion Card	Apple Computer	X		X	
Apple IIGS Memory Expansion Card	Apple Computer	Z		X	
Apple 3.5 Drive	Apple Computer	Z	Z	X	
Apple Color™ RGB Color Monitor	Apple Computer	X		X	
Apple ColorMonitor 100	Apple Computer	X		Z	
Apple ColorMonitor IIc	Apple Computer	X	X	X	Not recommended for new graphics mode on IIGS
Apple ColorMonitor IIe	Apple Computer	X		X	
Apple Color Plotter	Apple Computer	X		X	
Apple Daisy Wheel Printer	Apple Computer	X		X	
Apple Dot Matrix Printer	Apple Computer	X		X	
Apple Flat Panel Display	Apple Computer		X	Z	Connectors incompatible on IIGS
Apple Graphics Tablet	Apple Computer	X		X	
Apple Hand Controller II	Apple Computer	X		X	
Apple Joystick II	Apple Computer	X		X	
Apple Modem 300/1200	Apple Computer	X		X	
Apple Monitor II (Monochrome)	Apple Computer	X		X	
Apple Mouse	Apple Computer	X		X	
Apple Mouse Card	Apple Computer	X		X	
Apple Numeric Keypad IIe	Apple Computer	X		X	For IIGS, can be used with upgraded IIe only
Apple Personal Modem	Apple Computer	X		X	
Auto Dial 212A Modem	U.S. Robotics	X		X	
BusinessCard (Serial/Clock)	Street Electronics	X		Z	Multifunction, not compatible

H Hardware works; accompanying software needs update
N Works in Normal speed only
X Product works
Z Product doesn't work

Product	Company	IIe	IIc	IIGS	Comments
Cat 300 Baud Modem	Novation	X		X	
Centronics Parallel Printer	Centronics	X		X	
Champion Parallel Card	Cirtech	X		X	
Constellation II Network	Corvus	X		X	
Corvus 20 Mb Hard Disk	Corvus	X		N	
Corvus Multiplexer Network	Corvus	X		X	
Decillionix DX-1	Decillionix	X		X	
Disk Controller: Disk II®	Apple Computer	X		X	
Disk Controller: DuoDisk®	Apple Computer	X		X	Remove resistors from drive for IIGs*
Disk Controller: ProFile™	Apple Computer	X		Z	Card is now incompatible*
Disk Controller: UniDisk™ 3.5	Apple Computer	X		X	
Disk II	Apple Computer	X		X	
DuoDisk	Apple Computer	X		X	Remove one resistor from drive for IIGs*
Echo +	Street Electronics	X		N	
Elite One 5.25	Rana Systems	X		X	
Flipster Ram Card	Cirtech	X		Z	
Fourth Dimension 5.25	Fourth Dimension	X		X	
FX-80 Dot Matrix Printer	Epson	X		X	
Gold Card 192 K	Digital Research	X		X	
Graph Writer Printer Card	MPC	X		X	
Grappler + Parallel Interface	Orange Micro	X		X	
Grappler + Serial Interface	Orange Micro	X		X	
Grappler Serial Interface	Orange Micro	X		X	
Greengate Digital Sampler	Greengate	X		Z	
Greengate MIDI Interface	Greengate	X		Z	
Hitachi analog RGB	Hitachi			X	Product designed for Apple IIGs only
Hitachi VM 129 U Monitor	Hitachi	X		X	
HR 15 Dot Matrix Printer	Brother	X		X	
IEEE-488 Card	Apple Computer	X		X	
ImageWriter™ Wide Carriage Printer	Apple Computer	X		X	
ImageWriter II	Apple Computer	X		X	
ImageWriter (Standard)	Apple Computer	X		X	
IQS 401LX Spectrum Analyzer	IQS	X		X	40 column compatible
Keys-1Parallel Interface Card	Syntauri	X		X	Rev.B
Keys-1Parallel Interface Card	Syntauri	X		X	Rev.C
Koala + Pad	Koala Technologies	X		X	Poor response to buttons in Fast speed
Kurzweil Reading Machine	Kurzweil	X		X	
Legend E Card	Legend Industries	X		Z	No AUX slot in IIGS
Lis'ner 1000 Voice Recog.	MicroMint	X		X	2.0
Lobo 5.25 Disk Drive	Lobo	X		X	
M & R Enterprises Monitor	M & R Enterprises	X		X	
MD212 Modem	Ven-Tel	X		X	
MD212-3E Modem	Ven-Tel	X		X	
Micom Instalink 460 Modem	Micom	X		X	
Micro Music DAC Board	Temporal Acuity Products	X		N	
Micromodem IIe	Hayes	X		X	40 column compatible
Microsoft Softcard II System	Microsoft	X		H	
Microstore Hard Disk	AST	X		X	
Microtek Parallel Interface	Microtek	X		X	
Mikron ADC 4000 (RS232)	Mikron	X		X	
Model A-2 Modem	Micro SCI	X		X	
Mountain Music Card	Mountain Computer	X		X	Works with Metatrack 5.0
MultiRam IIe	Checkmate Tech.	X		Z	
Muppet Learning Keys	Koala Techologies	X		X	
MX-80 F/T Dot Matrix Printer	Epson	X		X	
Novation Cat Modem	Novation	X		X	
Novation Series 4000 Modem	Novation	X		X	
Orange Parallel Card	Orange Micro	X		X	
Panasonic Omni Monitor	Panasonic	X		X	
Parallel Interface Card	Apple Computer	X		X	
Passport MIDI Interface Card	Passport Designs	X		H	MasterTracks 5.0 doesn't work
Passport MIDI Interface Card	Passport Designs	X		H	With Tape Synch
PJ-1080A Printer	Canon	X		X	
PKASO/U Interface	PKASO/ U Interactive Struc.	X		X	
Power Pad	Chalk Board	X		X	
Print It!	Texprint	X		Z	
ProFile 10 Mb (Hard Disk)	Apple Computer	X		X	ProFile cards are not compatible*
ProFile 5 Mb (Hard Disk)	Apple Computer	X		X	ProFile cards are not compatible*
QC 10 (Hard Disk)	Quark	X		X	
Quick 20 (Hard Disk)	CMC Computer System	X		X	
Qume Sprint 5 Printer	Qume Corp.	X		X	
Ramworks	Applied Engineering	X		Z	
RX-80 Printer	Epson	X		X	
Sanyo DM5012 CX Monitor	Sanyo	X		X	
Sanyo VM 4209 Monitor	Sanyo	X		X	
Scribe® Printer	Apple Computer	X		X	
Seiko Watch-Data Terminal	CPU	X		X	
Sider II - 10 Mb (Hard Disk)	First Class Peripherals	X		X	
Sider II - 20 Mb (Hard Disk)	First Class Peripherals	X		X	
Silentype® Printer	Apple Computer	X		X	
SmartHome	Cyberlynx	X		X	
Smartmodem 2400	Hayes	X		X	
Softcard CATS Rev.A	Microsoft	X		X	
Super Serial Card	Apple Computer	X		X	
Super Serial Interface 7711	Apple Computer	X		X	
Swyftcard	Information Appliance	X		X	
Syntech MIDI Interface Card	Syntech	X		H	
Syntech MIDI Interface Card	Syntech	X		H	With Tape Synch
Telmac LDD-1 Modem	Telmac	X		X	
Telmac Modem	Telmac	X		X	
The CP/M Card Rev. B	ALS / Digital Research	X		Z	
Think JET Printer	Hewlett-Packard	X		X	
Thunderclock Plus	Thunderware	X		X	
U.S. Robotics Password Modem	U.S. Robotics	X		X	
Ufonic Voice System	Borg Warner	X		X	For the IIGS, card fits only upgraded IIe cases
UltraTerm Video Display	Videx	X		Z	
UniDisk 3.5	Apple Computer	X	X	X	
UniDisk (5.25)	Apple Computer	X	X	X	
Volksmodem	Volksmodem	X		X	
Z-Engine	Advanced Logic Systems	X		N	
Z80 Plus	Applied Engineering	X		N	

COME TO THE CHRISTMAS PARTY !!

- 50¢ DISK COPIES FROM THE Pi LIBRARY
- DOOR PRIZES
- FREE POP AND COOKIES
- SPECIAL DEAL ON PRINT SHOP GRAPHICS DISKS !!!
- LOTS OF FUN, BRING THE FAMILY

Pi CHRISTMAS PARTY

DEC. 4TH · School of MINES

6:00 PM - 10:PM

THE SEED

denver apple pi

P.O. Box 17467
Denver, CO 80217

First Class
U.S. POSTAGE
PAID
DENVER, CO
PERMIT- 3182